Caught in the honeypot

Brett Glass. **Boardwatch**. Golden: Jan 2000. Vol. 14, Iss. 1; pg. 114, 2 pgs

Abstract (Summary)
One of the most unfortunate problems network administrators face is that most of the time they will not find out that a cracker is running amok until after the fact. It is better, of course, to catch a would-be intruder before he or she gets in. One of the best ways to accomplish this is a honeypot - a decoy computer or program that looks as if it might be a point of entry, but is, in fact, a blind alley. While the intruder thinks he will be able to skulk around undetected, his every move is logged and traced; he will quickly wind up in trouble without actually penetrating the system. How to set up a honeypot is discussed.

Full Text (1735 words)

"Suppose," [Pooh] said to Piglet, "you wanted to catch me, how would you do it?"

"Well," said Piglet, I should do it like this. I should make a Trap, and I should put a Jar of Honey in the Trap, and you would smell it, and you would go in after it, and - "

"And I would go in after it," said Pooh excitedly, "only very carefully so as not to hurt myself, and I would get to the Jar of Honey, and I should lick round the edges first of all, pretending that there wasn't any more, you know, and then I should walk away and think about it a little, and then I should come back and start licking in the middle of the jar, and then

- A. A. Milne

One of the most unfortunate problems you'll face as a network administrator is that most of the time you will not find out that a cracker is running amok until after the fact sometimes long after the fact.

It's better, of course, to catch a would-be intruder before he or she gets in. But what's the best way to do this? Network monitors can sometimes help, as can watching one's logs. However, one of the best ways, by most accounts, is a "honeypot" a decoy computer or program that looks as if it might be a point of entry, but is, in fact, a blind alley.

What does a honeypot "look like" to a hacker bent on mischief? It might masquerade as an FTP server with known security holes, an old version of QPopper, a vulnerable CGI script on a Web server or even a Telnet daemon that seems to let an intruder break in by guessing passwords. The term "honeypot" - a euphemism for "chamberpot" - is apt.

While the intruder thinks he will be able to skulk around undetected, his every move is logged and traced; he'll quickly wind up in ... er ... deep excrement without actually penetrating the system. And if there are real security holes in the system or network, it will take the intruder longer to find them because he will be distracted and/or exposed by the bogus ones.

There's no record of when or where the first honeypot was implemented. However, they are reputed to have been used by Digital Equipment Corporation back when the Internet was still called the ARPANet - as a way of securing its Internet-connected computers against corporate espionage or tampering by college students. As the Net has grown, most major corporations, and many minor ones, have begun to use them, as well. So, regardless of other techniques you use to protect your network, it's a fine idea to set up a honeypot or two. But how to get started?

LAYING A TRAP

First and foremost, it's important that the system you set up as a honeypot really be a blind alley - not a real route to compromising your network. So, perversely, the system you set up to look like a hole in your network must be the securest machine you can configure.

Start with the latest version of a reputable OS - I recommend FreeBSD or OpenBSD - and lock it down tight. Shut down all unnecessary Internet services. Remove all extraneous user names, run all daemons in "sandboxes" (that is, as unprivileged users) and set the system up to log and report everything. (Some administrators prefer to do remote logging rather than logging attacks on the honeypot system itself, just in case that system really is compromised. My personal philosophy, however, is that allowing an attacker to actually get control of the system is too dangerous.)

I configure my honeypots in such a paranoid way that it almost would not be practical to use them for any real work. It may be a good idea to put your honeypot outside the firewall router that protects the rest of your site - or even put it on a separate interface on the firewall. That way, an attacker who subverts it will have additional barriers to overcome before getting the run of your network.

Next comes the fun part: setting the trap. This can be done in many ways, depending upon your levels of competence, chutzpah and programming skill. If you're not super-confident of your programming skills, you may find that one easy way is to start by leveraging someone else's work.

Search the Net for the word "honeypot" using any of the major search engines; you'll find plenty of "toolkits" - commercial and free - which can transform a computer into a workable honeypot. Lance Spitzner's "To Build A Honeypot" page at www .enteract-com/-lspitz/honey pot.html is a good starting point, as are the Web sites of Network Associates (see www.nai.com/asp -set/products/tns/ccsting _intro.asp) and Fred Cohen and Associates (see http://all.net /dtk/dtk.html).

As you become more comfortable with the package you're using, you'll probably want to diverge from the "cut and dried" material in the kit and roll your own.

ROLLING YOUR OWN

If you're braver, or if you have more security experience, you may want to craft your own trap from the start. Here are some ideas to get you going.

One of the easiest ways to get a would-be cracker's attention is to make the honeypot look like a machine that's running an older, crustier, less secure operating system and equally old utilities. Change the system's "message of the day" (motd for short on Unix systems) to make it appear to be running an old OS version - preferably one that uses a different brand of CPU than is actually present on the machine. Do the same for utilities such as FTP, a POP server and the Web server.

Name the system something that implies it is of great importance to your network and contains valuable data for example, "crediteardorders.myisp.com" or "accountinfo.myisp.com" - and add this name to your forward and reverse DNS zone files. (When the name shows up in a scan of your IP address range, crackers will be misdirected away from your production machines to the honeypot.)

Configure the machine as a Web server and put up a few pages with simple forms - clearly insecure ones - that ask for credit card numbers and other personal information. (No legitimate user will ever have any reason to visit the site, but an intruder will note these forms and assume such information is stored there.)

Visit the CERT database (www. cert. org), and pick a few system vulnerabilities to emulate. Depending on your level of coding skill, you should be able to write some simple, unprivileged daemons which look like a compromisable FTP server or mail server in an hour or two. (You don't have to implement much of the functionality of the actual program; the key is getting the initial login message correct.)

If you're feeling really crafty, You might try to lead the intruder down the garden path by making it appear that he or she has gotten into the system and "broken root." But be careful: such charades often backfire. A truly good cracker will become suspicious if breaking into the system is too easy - and the farther the attacker is allowed to go, the more likely it is he or she will actually be able to get control of the system.

IDENTIFYING INTRUDERS

Let's suppose you've configured your honeypot, put it out on the Net and waited. On today's Internet, it probably won't take long - my guess would be less than a day or two - before you get a "live one" - someone trying to break in.

When this happens, the first thing to do is cast aside romantic notions of cops and robbers, secret agents and intrigue and be practical. The odds are overwhelming that this is not a master hacker, but rather a "skript kiddie" testing the latest exploit or scanning program he's downloaded from the Net. And because the Internet is global, a large percentage of attacks come from overseas - making apprehension of the culprit difficult or even impossible. So, don't overreact. Do set your honeypot up to capture the intruder's IP address and trace it back to the source automatically if possible. (Your queries could alert the intruder to the fact that he is being watched, but if it does, it's probably a good thing; it takes far less time and effort to scare a cracker off than to track him down.)

You may be disappointed to discover that the intruder is good at covering his tracks - for example, he might be coming through an open Wingate or some other previously compromised system - but at least you'll be able to notify the owner of that system that hackers are exploiting it. If attacks seem to be coming from the same addresses repeatedly, and you can't get the owner of the addresses to take action, block the relevant IP addresses at your firewall.

PART OF THE PICTURE

Honeypots are, of course, only part of the security picture. Be sure to employ other measures - including network monitoring and routine scans of your own systems - to detect intruders and vulnerabilities. Still, having a honeypot or two on your network is fun and useful. If nothing else, they'll give you, as an administrator, a good sense of who's trying to break into your network and how. And, as they say, forewarned is forearmed.

'SPLOIT OF THE MONTH: HOLES IN BIND 8

This month's featured exploit is actually a set of exploits recently found in BIND, the Berkeley Internet Name Daemon. Since nearly every site on the Net runs BIND, the holes described by CERT at www.cert.org/ad vigories/CA-99 -14 -bind. html - which range from denial of service attacks to root compromises are of great concern. Check with the vendor of your operating system for appropriate patches.*

[Author Affiliation]
Brett Glass has more than 20 years of experience designing, building, writing about and crashtesting computer hardware and software. Glass obtained his Bachelor of Science degree in Electrical Engineering from the Case Institute of Technology and his MSEE from Stanford. When he's not writing, consulting, speaking or cruising the Web in search of adventure, he may be playing the Ashbory bass, teaching Internet courses for LARIAT (Laramie's community network and Internet users' group), cooking
up a storm or enjoying spicy ethnic food.
To reach Glass, go to www.brettglass. COM/ mailbrett.html.

Indexing (document details)

Subjects:	Computer security, Systems design, Guidelines, Hackers
Locations:	United States, US
Author(s):	Brett Glass
Document types:	Instructional
Publication title:	Boardwatch. Golden: Jan 2000. Vol. 14, Iss. 1; pg. 114, 2 pgs
Source type:	Periodical
ISSN:	10542760
ProQuest document ID:	47669289
Text Word Count	1735
Document URL:	http://proquest.umi.com/pqdweb?did=47669289&sid=3&Fmt=3&clie ntId=19649&RQT=309&VName=PQD